Learning Deformations with Parallel Transport

Donglai Wei, Dahua Lin, and John Fisher III

CSAIL, MIT

Abstract. Many vision problems, such as object recognition and image synthesis, are greatly impacted by deformation of objects. In this paper, we develop a deformation model based on Lie algebraic analysis. This work aims to provide a generative model that explicitly decouples deformation from appearance, which is fundamentally different from the prior work that focuses on deformation-resilient features or metrics. Specifically, the deformation group for each object can be characterized by a set of Lie algebraic basis. Such basis for different objects are related via parallel transport. Exploiting the parallel transport relations, we formulate an optimization problem, and derive an algorithm that jointly estimates the deformation basis for a class of objects, given a set of images resulted from the action of the deformations. We test the proposed model empirically on both character recognition and face synthesis.

1 Introduction

The changes in shapes of objects, often referred to as *deformations*, are widely observed in computer vision data. In many problems, such as object recognition, the performance can be greatly influenced by deformations. Whereas past decades have seen tremendous efforts devoted to developing features and classifiers that are resilient to shape variations, the modeling of deformations has not been extensively explored. In this paper, we focus on modeling deformations, aiming to develop a method that can decouple deformation from object appearance.

We provide a brief review of existing approaches to deformation analysis in next section. Careful examination of these approaches suggests that they are limited in several aspects: (1) Manifold-based methods [1, 2] are popular in image modeling. Such methods, though partly capturing shape variations, are usually not very effective in modeling deformations. The key issue here is the lack of a mechanism that can decouple the effects of deformations and other factors that contribute to the variations of appearance. (2) The methods for deformation-resilient metrics [3, 4] aim to suppress the influence of deformation on discriminative performance, which again does not offer an explicit deformation model. (3) Other work that explicitly takes deformations into consideration [5–7] has a narrow focus on individual local tangent spaces, neglecting the relations between them. As we will show, there are significant dependencies between the different tangent spaces of the deformation manifolds, which, if appropriately exploited, contribute greatly to learning a model of deformation.

In this paper, we propose a new approach to deformation modeling, where each observed image is considered to be generated by deforming an object template. In typical images, the deformation of an object usually exhibit some regular patterns. This observation leads to the belief that deformations of an object can be approximated by a low-dimensional Lie group. In general, a Lie group is uniquely associated with a vector space, called the Lie algebra, which can be characterized by a set of bases. Intuitively, each base vector in this space can be considered as a base deformation pattern, and any deformation in the group can be expressed as a linear combination of these bases with the Lie algebraic characterization. Generally, a different Lie algebra is associated with a different object template. For object templates that represent different poses of an object, the associated Lie algebras are related to each other via the parallel transport property. Specifically, the Lie algebra for one object template is a transported version of the one for others. The fact that parallel transport is covariant with geometric transformation ensures the consistency of this relation.

Consequently, with the Lie algebraic characterization, the problem of learning deformations reduce to the one of estimating the deformation bases for different object templates. Here, we formulate an optimization problem for estimating these bases from a given set of observed images. In this formulation, two levels of relations are exploited:

1. Observed images are closed to the deformation orbits, *i.e.* the manifold is comprised of all deformed versions of the templates.
2. The bases associated with different templates are constrained by the parallel transport relations.

The use of the first relation, which explicitly incorporates deformation into the generative process of an image, clearly sets this work apart from the large amount of prior work (*e.g.* those on image manifold learning) that directly model the image space. Additionally, the use of the parallel transport relation further distinguishes the proposed approach from the methods which focus on local neighborhoods only.

The remainder of this paper is organized as follows. Section 3 reviews existing theoretical results on deformations. The emphasis is particularly placed on the Lie algebraic characterization and parallel transport. Section 4 formulates the optimization algorithm for estimating the deformation model from observed images. Empirical results are presented in section 5, where we compare the proposed method with related methods on character recognition and synthesis, as well as face reconstruction. Discussion of the method and results is provided in section 6.

2 Related Work

We first briefly review previous work on deformations, which roughly fall into two categories. The first category of methods focuses on estimating global affine transformation. Frey and Jojic [8–10] proposes a mixture model, where the space

of affine transforms is discretized, and an indicator is used to choose a specific transform in generating each image. Miller *et. al* [11] proposed a nonparametric probabilistic model, which estimates the global affine transforms by gradually aligning the images, using a gradient descent method referred to as "congealing".

The second category takes into account non-rigid deformations that can lead to changes in shapes. Cootes *et. al* [12, 13] proposed the active appearance model for object alignment, where the deformation is represented via the displacement of pre-specified control points. In addition, approaches by directly matching local descriptors are also widely used. Belongie et al. [3] developed a direct matching method using local shape context based on statistics of edges. Keyser et al. [4] developed an Image Distortion Model (IDM) [4], which pursues a dense match of local patches between two images as a representation of the deformation. Though simple, this method leads to substantial improvement on character recognition, providing significant evidence as to the important role of local deformations in object recognition. The pioneering work by Tenenbaum etal [1] and Roweis and Saul [2] initiated a large amount of work that directly models the image manifold via embeddings it into local low-dimensional spaces.

While deformation information is made use of in building object metrics in the work mentioned above, these methods do not establish an explicit model of deformations. Recently, new models have been proposed to address this issue. Simard et al. [5] considered the manifold of deformations, approximating it via local tangent spaces. In this work, the basis of these tangent spaces are handcrafted, with some apparent deformation patterns taken into account (*e.g.* rotation and changes of thickness). However, some subtle variations of shapes are difficult to capture via manually devised patterns. Another drawback of this method lies in the introduction of tangent spaces for all training samples, incurring unnecessary computational costs in both training and testing phases when the samples are dense. Hastie and Simard [6, 7] improve upon this method by grouping nearby samples into clusters and deriving the tangent basis via learning. However, learning is performed independently for each tangent space, utilizing only the samples within a local neighborhood. This makes it potentially difficult to obtain reliable estimations.

3 The Theory of Deformation

Generally, the shape and size of an non-rigid object can change over time. Such a change is often referred to as a *deformation*, which is ubiquitous in vision problems. In this paper, we focus on the two-dimensional image space, where a deformation can be formalized as a *diffeomorphic transform* on the image plane.

3.1 Lie Group and Lie Algebra

Deformations typically observed in vision problems are a subset of all diffeomorphic transforms, which we assume constitute a Lie group of dimension K. A Lie group G is a finite-dimensional manifold with an algebraic group structure, meaning that it has the following properties:

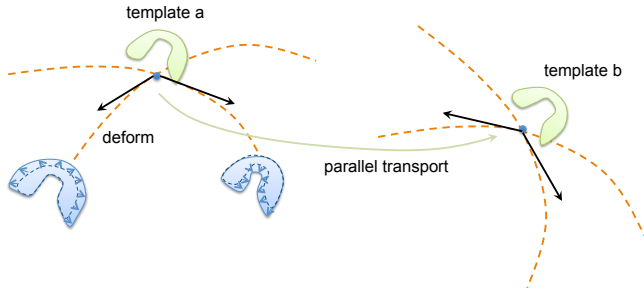

Fig. 1. This figure illustrates the Lie algebraic characterization of deformation groups. Here, starting from an object template, a deformed image can be generated following a velocity field, which can be expressed as a linear combination of some basic patterns (captured by the Lie algebraic basis). The basis associated with different object templates are related by parallel transport.

1. The identity transform is in G.
2. If T_1 and T_2 are both in G, then the composition $T_1 \circ T_2$ is also in G.
3. For each transform $T \in G$, the inverse transform T^{-1} also exists in G.

The Lie group G is associated with a Lie algebra $\mathfrak{g}$, a vector space of dimension K. Each vector $V \in \mathfrak{g}$ is a velocity field and corresponds uniquely to a transform $T \in G$ via the exponentiation mapping as below

$$T = \exp(V). \tag{1}$$

Here, V is called the *Lie algebraic representation* of T.

The relations between a Lie group G and its associated Lie algebra $\mathfrak{g}$ can be described through the construction of a continuous transformation process. Let $V \in \mathfrak{g}$, then for every $t > 0$, $T_t = \exp(tV)$ is a transform. Hence, the function below defines a trajectory on the image plane.

$$\mathbf{x}(t) = T_t \mathbf{x}_0 = \exp(tV)\mathbf{x}_0. \tag{2}$$

Intuitively, this trajectory can be generated through a continuous transformation process described as follows. Consider a particle starting from $\mathbf{x}_0$. If the particle travels across the image plane, passing through each location $\mathbf{x}(t)$ with velocity $V(\mathbf{x}(t))$, the resultant trajectory is then given by

$$\mathbf{x}(t) = \mathbf{x}_0 + \int_{\tau=0}^{t} V(\mathbf{x}(\tau))d\tau. \tag{3}$$

This provides a detailed characterization of the trajectory defined in Eq.(2), namely $\exp(tV)\mathbf{x}_0$. Therefore, the transform $\exp(tV)$ can be understood as an operation that sends each point to move for time t, following the velocity field V. Equivalently, the trajectory is characterized by the differential equation below

$$\frac{d\mathbf{x}(t)}{dt} = V(\mathbf{x}(t)). \tag{4}$$

Given a basis of $\mathfrak{g}$, denoted by $\mathcal{B} = (B_1, \ldots, B_K)$, each Lie algebraic vector $V \in \mathfrak{g}$ can be expressed as a linear combination as $V = \sum_{k=1}^{K} \alpha^k B_k$. As illustrated by Figure 1, each base vector of $\mathfrak{g}$ reflects a basic deformation pattern, and all deformations in G are combinations of such base patterns. The Lie algebraic characterization provides a representation, where such combinations can be done via linear operations, great simplifying the modeling and estimation.

3.2 The Action on Images

A deformation $T \in G$ can act on an image by moving the locations of its pixels. Let I be an image. Applying T to I results in an deformed image $T \circ I$, given by

$$(T \circ I)(\mathbf{x}) = I(T^{-1}\mathbf{x}). \tag{5}$$

This means that the pixel value of $T \circ I$ at $\mathbf{x}$ equals that of I at $T^{-1}\mathbf{x}$. Let $V \in \mathfrak{g}$. Applying a continuous transform process $\exp(tV)$ to the image I yields a continuous sequence of images, as

$$I_t(\mathbf{x}) = (\exp(tV) \circ I)(\mathbf{x}) = I(\exp(-tV)\mathbf{x}). \tag{6}$$

Taking the derivative *w.r.t.* t, we get

$$\left.\frac{dI_t(\mathbf{x})}{dt}\right|_{t=0} = -V(\mathbf{x})^T \nabla I(\mathbf{x}) \triangleq (V \circ I)(\mathbf{x}). \tag{7}$$

Here, $V \circ I$ denotes the *action of* V *on* I, which produces a scalar map, whose value at $\mathbf{x}$ equals the negated inner product between the velocity $V(\mathbf{x})$ and the image gradient $\nabla I(\mathbf{x})$. Clearly, the action of V is a linear operation on I.

Given a basis $\mathcal{B}$, we can write V in form of a linear combination as $V = \sum_{k=1}^{K} \alpha^k B_k$. Consequently, we can rewrite Eq.(7) into

$$\left.\frac{dI_t(\mathbf{x})}{dt}\right|_{t=0} = \sum_{k=1}^{K} \alpha^k (B_k \circ I)(\mathbf{x}). \tag{8}$$

This equation establishes the linear isomorphism between the Lie algebraic representation and the image changes due to deformation. Specifically, the infinitesimal changes generated by a deformation $V = \sum_{k=1}^{K} \alpha^k B_k$ can be expressed as a linear combination of the "base changes" – those generated by the base deformations – with the same set of coefficients $\alpha_1, \ldots, \alpha_K$.

3.3 Parallel Transport

In general, a deformation group is associated with a specific object, which can not be directly applied to a different object (*e.g.* a transformed version of the object). However, one can adapt a deformation group via the *parallel transport* of the associated Lie algebra, enabling its application to different objects.

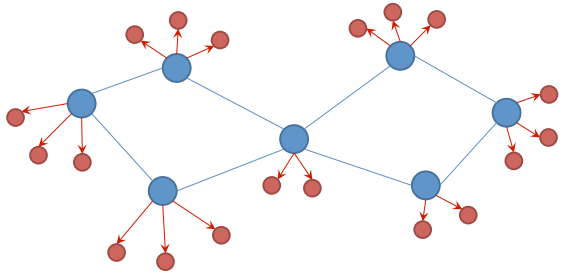

Fig. 2. This figure shows the two-level formulation of the proposed optimization formulation. At the first level, each center image (depicted by blue circles) are connected to all its neighbors (red circles) within the same cluster, and at the second level, different centers are connected via parallel transport constraints.

Consider an object being deformed, which are observed from two different views. The point at $\mathbf{x}$ from the first view is transformed to $\mathbf{x}' = T\mathbf{x}$ from the second view. Suppose this point has velocity $\mathbf{v}$ at $t = 0$ from the first view, then *what is the velocity of the corresponding point, i.e.* $T\mathbf{x}$*, from the second view?* The derivation below shows the answer:

$$\mathbf{v}' := \lim_{\delta t \to 0} \frac{T(\mathbf{x} + \mathbf{v}\delta t) - T(\mathbf{x})}{\delta t} = \mathbf{J}_T(\mathbf{x})\mathbf{v}. \tag{9}$$

Here, $\mathbf{J}_T(\mathbf{x})$ is the Jacobian matrix of T at $\mathbf{x}$. Here $\mathbf{v}'$ is called the *parallel transport* of $\mathbf{v}$ *w.r.t.* the transform T. The parallel transport can be applied to an entire velocity field V, resulting in a new velocity field $T \bullet V$, given by

$$(T \bullet V)(T\mathbf{x}) = \mathbf{J}_T(\mathbf{x})V(\mathbf{x}). \tag{10}$$

The parallel transports are *covariant* with the inducing transforms, meaning that they satisfy two properties below: (1) the parallel transport induced by an identity transform in itself is an identity, and (2) the parallel transport induced by a composition of two transforms equals the composition of the transports respectively induced, as $(T_2T_1) \bullet V = T_2 \bullet (T_1 \bullet V)$.

4 Model Estimation Algorithm

In this section, we formulate an optimization problem to estimate the deformation groups for a specific class of objects, given a set of images, and thereon derive an algorithm that jointly solves the basis of the deformation groups and the Lie algebraic coefficients for the training samples.

4.1 Two-Level Formulation

Given a set of n images, we first group them into m clusters, using K-medoid, where each cluster has a *center image*. The number of clusters m is chosen

via cross validation, such that all samples within a cluster are close enough to the corresponding center. Suppose the i-th cluster contains n_i samples. For this cluster, we use $I_{i,0}$ to denote the center image of this, and $I_{i,j}$ (with $j = 1, \ldots, n_i$) to the j-th non-center image. Here, we consider each center image as the representation of the canonical shape of an object, and other images in the same cluster as generated by deforming the center image.

As discussed in previous section, a deformation group can be characterized by a Lie algebra. With the Lie algebraic characterization the problem of learning the deformation groups reduces to the one of estimating the Lie algebraic basis for each cluster. Here, we denote the basis for the i-th cluster by $\mathcal{B}_i = (B_{i,1}, \ldots, B_{i,K})$. To estimate these basis, we formulate an optimization problem, of which the objective function comprises two levels of terms, as shown in Figure 2.

Within-cluster Level. Applying the deformation group characterized by the Lie algebraic basis $\mathcal{B}_i$ to the image $I_{i,0}$ yields a K-dimensional manifold comprised of all the deformed images, denoted by $G(\mathcal{B}_i) \circ I_{i,0}$, as

$$G(\mathcal{B}_i) \circ I_{i,0} \triangleq \{\exp(V) \circ I_{i,0} : V \in \mathfrak{g}(\mathcal{B}_i)\}. \tag{11}$$

Here, $\mathfrak{g}(\mathcal{B}_i)$ denotes the Lie algebraic space spanned by $\mathcal{B}_i$. With the assumption that $I_{i,j}$ is generated by deforming $I_{i,0}$, we expect that the $I_{i,j}$ is close to $G(\mathcal{B}_i) \circ I_{i,0}$. Particularly, the distance from $I_{i,j}$ to $G(\mathcal{B}_i) \circ I_{i,0}$ is given by

$$\operatorname{dist}(I_{i,j}, G(\mathcal{B}_i) \circ I_{i,0}) = \min_{\boldsymbol{\alpha}} \left\| I_{i,j} - \exp\left(\sum_{k=1}^{K} \alpha^k B_{i,k}\right) \circ I_{i,0} \right\|. \tag{12}$$

When the deformed image $I_{i,j}$ is close to the center $I_{i,0}$, the coefficients are small. Consequently, by Eq.(8), we can approximately write

$$\exp\left(\sum_{k=1}^{K} \alpha^k B_{i,k}\right) \circ I_{i,0} \simeq I_{i,0} + \sum_{k=1}^{K} \alpha^k (B_{i,k} \circ I_{i,0}). \tag{13}$$

As a result, we have

$$\begin{aligned} \operatorname{dist}(I_{i,j}, G(\mathcal{B}_i) \circ I_{i,0})^2 &\simeq \min_{\boldsymbol{\alpha}} \left\| (I_{i,j} - I_{i,0}) - \sum_{k=1}^{K} \alpha^k (B_{i,k} \circ I_{i,0}) \right\|^2 \\ &= \min_{\boldsymbol{\alpha}} \sum_{\mathbf{x} \in \mathcal{D}} \left((I_{i,j}(\mathbf{x}) - I_{i,0}(\mathbf{x})) + \sum_{k=1}^{K} \alpha^k B_{i,k}(\mathbf{x})^T \nabla I_{i,0}(\mathbf{x}) \right)^2. \end{aligned} \tag{14}$$

Here, $\mathcal{D}$ is the set of all observable pixel locations. For convenience, we define

$$Q_{ij}(\mathcal{B}_i, \boldsymbol{\alpha}_{i,j}) = \left\| (I_{i,j} - I_{i,0}) - \sum_{k=1}^{K} \alpha_{i,j}^k (B_{i,k} \circ I_{i,0}) \right\|^2. \tag{15}$$

Note that Q_{ij} is quadratic *w.r.t.* $\boldsymbol{\alpha}_{i,j}$. Hence, the optimal coefficients that yield the minimum (approximate) distance can be readily solved, given $\mathcal{B}_i$.

Inter-Cluster Level. The basis associated with different groups are related to each other via parallel transport. Specifically, we establish a higher-level network between cluster centers, where each center image is connected to several *neighboring centers*, *i.e.* other centers that are not too far from it, such that the optical flow between them can be reliably estimated.

For each pair of neighboring centers $I_{i,0}$ and $I_{i',0}$, we estimate the dense correspondence between them $T_{ii'}$ and $T_{i'i}$, using an optical flow algorithm [14]. Ideally, we would expect the basis $\mathcal{B}_i$ to be the transported version of $\mathcal{B}_{i'}$ *w.r.t.* the transform $T_{i'i} = T_{ii'}^{-1}$, *i.e.* $B_{i,k} = T_{ii'}^{-1} \bullet B_{i',k}$, and vice versa. As some errors may arise in optical flow estimation, we use the quadratic term as follows to penalize the deviation from this relation:

$$H_{ii'}(\mathcal{B}_i, \mathcal{B}_{i'}) = \sum_{k=1}^{K} \|B_{ik} - T_{ii'}^{-1} \bullet B_{i',k}\|^2 \tag{16}$$

Here, we have

$$\|B_{ik} - T_{ii'}^{-1} \bullet B_{i,k}\|^2 = \sum_{\mathbf{x} \in \mathcal{D} \cap T_{ii'}(\mathcal{D})} \|B_{ik}(\mathbf{x}) - \mathbf{J}_{T_{ii'}}(T_{ii'}(\mathbf{x})) B_{i'k}(T_{ii'}(\mathbf{x}))\|. \tag{17}$$

Here, $T_{ii'}(\mathbf{x})$ is the location of the pixel on $I_{i',0}$ that corresponds to the pixel at $\mathbf{x}$ of $I_{i,0}$. In general, $T_{ii'}(\mathbf{x})$ does not yield integer coordinates. Under such circumstances, linear interpolation can be used to derive the values of $\mathbf{J}_{T_{ii'}}(T_{ii'}(\mathbf{x}))$ and $B_{ik}(T_{ii'}(\mathbf{x}))$. In addition, $\mathbf{x} \notin T_{ii'}(\mathcal{D})$ indicates that the pixel at $\mathbf{x}$ of $I_{i,0}$ is transformed outside of the observable region, and thus the corresponding term is not included.

Joint Formulation. Integrating the terms at both levels, we derive the joint objective function as follows.

$$L(\mathcal{B}) = \sum_{i=1}^{m} \sum_{j=1}^{n_i} \min_{\boldsymbol{\alpha}} Q_{ij}(\mathcal{B}_i, \boldsymbol{\alpha}) + \gamma \sum_{i=1}^{m} \sum_{i' \in \mathcal{N}_i} H_{ii'}(\mathcal{B}_i, \mathcal{B}'_i). \tag{18}$$

Here, $\mathcal{N}_i$ is a set consisting of the indices of $I_{i,0}$'s neighboring centers, and γ is a positive weight that controls the contribution of the parallel transport constraints. To minimize this function, we introduce an auxiliary function that involves $\boldsymbol{\alpha}_{i,j}$ as arguments:

$$L_{aux}(\mathcal{B}, \boldsymbol{\alpha}) = \sum_{i=1}^{m} \sum_{j=1}^{n_i} Q_{ij}(\mathcal{B}_i, \boldsymbol{\alpha}_{i,j}) + \gamma \sum_{i=1}^{m} \sum_{i' \in \mathcal{N}_i} H_{ii'}(\mathcal{B}_i, \mathcal{B}'_i). \tag{19}$$

Obviously, L_{aux} gives an upper bound of L and has

$$L(\mathcal{B}) = \min_{\alpha} L_{aux}(\mathcal{B}, \boldsymbol{\alpha}). \tag{20}$$

Consequently, $L(\mathcal{B})$ can be optimized by alternating the updates of $\boldsymbol{\alpha}$ and $\mathcal{B}$:

$$\hat{\boldsymbol{\alpha}}_{i,j}^{(t)} \leftarrow \underset{\boldsymbol{\alpha}}{\operatorname{argmin}}\ Q_{ij}(\mathcal{B}_i^{(t-1)}, \boldsymbol{\alpha}), \tag{21}$$

$$\hat{\mathcal{B}}_i^{(t)} \leftarrow \underset{\mathcal{B}}{\operatorname{argmin}} \sum_{j=1}^{n_i} Q_{ij}(\mathcal{B}, \boldsymbol{\alpha}_{i,j}^{(t)}) + \gamma \sum_{i' \in \mathcal{N}_i} H_{ii'}(\mathcal{B}_i, \mathcal{B}_{i'}). \tag{22}$$

Note that the value of $L_{aux}(\mathcal{B}, \boldsymbol{\alpha})$ decreases with each updating step. Particularly, the values of L and L_{aux} become equal each time when $\boldsymbol{\alpha}$ is updated to the optima, *i.e.* $L(\mathcal{B}^{(t)}) = L_{aux}(\mathcal{B}^{(t)}, \boldsymbol{\alpha}^{(t+1)})$.

4.2 Initialization

While L_{aux} is convex *w.r.t.* $\mathcal{B}$ and $\boldsymbol{\alpha}$ respectively, this is not a convex optimization problem jointly. Hence, appropriate initialization is crucial as to obtaining a reasonably good solution. Here, we describe a simple yet effective scheme to initialize the the basis $\mathcal{B}$.

We choose a particular cluster as the "standard cluster", and compute the optical flow [14] from the center of this standard cluster to the centers of other clusters. With these optical flows, we can warp the images in other clusters towards the standard one. For example, suppose $I_{1,0}$ is selected as the standard, and the optical flow from $I_{1,0}$ to $I_{2,0}$ is T_{12}, then we warp each image in the second cluster as $I'_{2,j} = T_{12}^{-1}(I_{2,j})$ for $j = 0, 1, \ldots, n_2$. In this way, for each non-standard cluster, we acquire a warped center as well as a set of warped images, which are considered as generated by deforming the warped center.

At the initialization stage, we assume that the standard cluster and the warped clusters share the same basis. To estimate this basis, we compute the optical flow fields from the standard center to other images in the standard cluster, and those from each warped center to other images in the corresponding cluster. All these flow fields can be roughly considered as residing near the space spanned by the shared basis. Therefore, the basis can be estimated by applying principal component analysis (PCA) to these optical flow fields pooled together. After the basis associated with the standard cluster is initialized, the bases for other clusters can be readily obtained via parallel transport.

5 Experiments

Given the learned prototypes and deformation bases, we now have a generative model for images. In this section, we apply our deformation model to two vision tasks: (1) handwritten character recognition with training sets of varying sizes, and (2) synthesis of digits and human faces. These experiments demonstrate the utility of the proposed method on both discriminative and generative tasks.

5.1 Handwritten Digit Recognition

On the popular MNIST [15] dataset, state-of-the-art algorithms can achieve very high accuracy (with error rates less than 0.5%). However, these methods usually rely on a large training set that densely cover all possible variations of each character. Actually, in such a training set, one can find very close matches to most testing samples. Consequently, a simple K-Nearest Neighbor (KNN) method [4] with a properly chosen metric suffices to achieve a very low error rate. However, a large data set is often cumbersome in practice, and methods relying on dense data set are difficult to generalize. In this experiment, we compare the proposed approach with several widely used algorithms in handwritten digit recognition. As we will see, the results obtained on training sets of varying sizes show that a structured deformation model can improve generalizability, making it possible to maintain an comparably effective model with much smaller number of prototypes. We compare four methods in the experiments:

1. **L2**: Find the nearest sample in terms of Euclidean distance in the feature space, and classify the testing sample to the resultant sample's class.
2. **TD**: Construct a tangent space for each training sample using a set of predefined bases, and find the closest tangent spaces to a given testing sample. This method is known as *Tangent Distance* [5].
3. **IDM**: This is a well-known method proposed by Keysers *et al.* [4], which was the best-performing method on MNIST. The basic idea is to divide an image into small patches, with each patch matched to the closest patch in a training image. The patch is allowed to move within a small window during the matching.
4. **DL-PT**: Use the proposed deformation model. With a learned model, a tangent space is associated with each prototype. The bases for different prototypes are different, but they are related through *parallel transport*.

In addition, two image features are considered: *pixel intensities (pix)* and *Sobel gradients (sob)*.

For each method, the training is performed on a training set of varying sizes, with the purpose of testing their generalizability. In particular, for our deformation model, the prototypes are found by K-medoid clustering, while the associated bases are jointly learned with the parallel transport relations taken into account. Here, the dimension of each tangent space (*i.e.* the size of the basis) is determined empirically (through PCA with 90% of the variation preserved in the principal subspace). Note that our focus here is to test the effectiveness of the deformation model instead of comparing classifiers. Hence, we use best-match strategy for each method. However, one can adapt more powerful classifiers, such as Support Vector Machine (SVM) and Convolutional Networks to further improve the recognition accuracy.

Figure 3 shows that when the training set is small, all methods make a considerable amount of errors, and the error rates decrease as the training set grows. We can see that with a structured deformation model, the error rate yielded by the proposed method clearly drops faster than that by other methods.

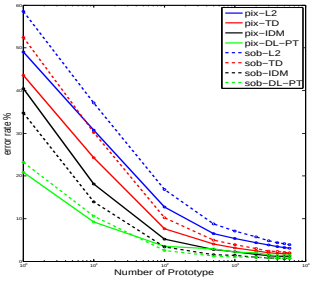

Fig. 3. Filled Line: Recognition error on MNIST dataset with pixel intensity as the feature; Dashed Line: Recognition error on MNIST dataset with Sobel gradients as the feature.

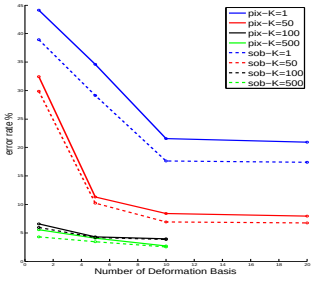

Fig. 4. Filled Line: Recognition error on MNIST dataset with pixel intensity as the feature; Dashed Line: Recognition error on MNIST dataset with Sobel gradients as the feature.

This is partly ascribed to the fact that statistical strength is shared among local models via parallel transport constraints.

In the second experiment, we vary both the number of prototypes and the number of deformation basis to investigate their influence in classification performance. In Figure 4, each colored curve corresponds to the error rate for a given number of prototypes and the x-axis of each point on the curve represents the number of deformation basis selected for each digit. As expected, increasing the number of local components (*i.e.* a prototype together with its tangent space) and the tangent space dimension generally leads to better classification performance. In addition, we notice that the performance becomes stable as the number of local components increases beyond a threshold (about 500), which is clearly much smaller than the size of the entire training set.

5.2 Image Synthesis

While many image synthesis experiments are designed to demonstrate super resolution results which are appealing from a human perceptual standpoint, our primary purpose is different. We instead wish to show that the learned basis, which is shared across the manifold, are meaningful from a geometric perspective.

Digit synthesis. Given the digit manifold learned from MNIST dataset, we synthesize new images of digits by generating samples from a randomly selected local component with random Lie algebraic coefficients. In particular, given a coefficient vector, a sample can be generated through integration along a geodesic on the learned manifold.

The top row of Figure 5 shows the sampled prototype of each digit in the first column and synthesized new digit images in the rest of each row. One can see that the synthesized images reflect local variations which the global affine

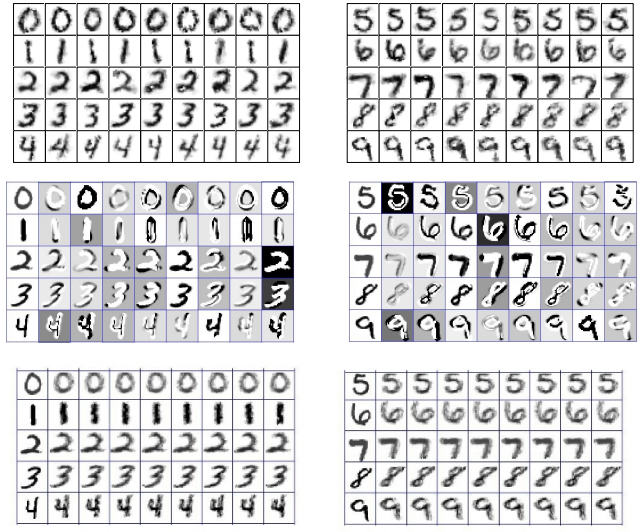

Fig. 5. Synthesized digits from the learned digit deformation manifold. The first digit in the row is the prototype; the rest are locally deformed from the prototype with a random coefficients of the learned basis

transforms are not able to explain. For example, for digit "2", we can see that there are some basis related to the size of the lower left circle of the digit. For comparison, we plot the synthesis results using Tangent Distance (TD) [5] (whose bases are pre-defined rather than being learned) in the middle row and that using IDM (averaging over the randomly shifted patches of the image) in the bottom row. Note that TD makes a combination of rigid transformation, hyperbolic deformations and intensity deformations to the prototype, while IDM only changes the detail of the prototype.

Face synthesis. In addition, we learn a face manifold using the face dataset of Brendan Frey, containing around $2,000$ grayscale images of size 20×28 in

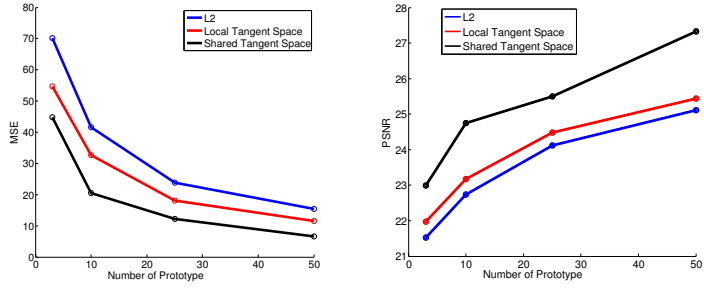

Fig. 6. Comparison of reconstruction performance on face synthesis.

different expressions and angles of view. For this experiment, we first learn the commonly shared basis to construct the manifold from $1,000$ sampled images. Then, for a separate set of randomly sampled 500 testing images, we try to see how close they can be projected onto the manifold. We tested three different algorithms for reconstruction: nearest training image in Euclidean metric, closest projection onto tangent spaces and closest projection onto our connected deformation manifold with shared basis. Again, we test our results with a varying number of prototypes. Figure 6 shows that, in terms of both Euclidean distance and PSNR ratio, the reconstruction from the manifold with a learned shared basis is consistently better than those learned independently from training examples. Note that the images are small and many reconstruction errors are not obviously perceivable. Consequently, we feel that a quantitative evaluation is more appropriate than showing the reconstructed faces.

6 Conclusion

We have presented a new method for manifold learning over images. The method is distinct from previous approaches in that the model explicitly incorporates a local Lie algebraic representation of deformations combined with a consistency relation derived from the parallel transport property. While previous methods consider local tangent spaces parameterized by a deformation basis, the methods of which we are aware utilize a hand crafted basis in contrast to the presented method which learns the basis. This process was enabled by exploiting the parallel transport property which imposes geometric consistency across local tangent spaces and effectively leverages the full training set (rather than local clusters) for learning properties of the deformation manifold. An efficient coordinate descent algorithm was presented along with a suggested initialization procedure. Empirical results demonstrating the utility of the methodology were presnted for hand written character recognition and synthesis as well as human face reconstruction.

Acknowledgement

D. Wei was partially supported by ONR MURI grant N00014-09-1-1051. D. Lin was partially supported by the Office of Naval Research Multidisciplinary Research Initiative (MURI) program, award N000141110688. J. Fisher was partially supported by DARPA award FA8650-11-1-7154.

References

1. Tenenbaum, J., Silva, V., Langford, J.: A global geometric framework for nonlinear dimensionality reduction. Science **290** (2000) 2319
2. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. Science **290** (2000) 2323

3. Belongie, S., Malik, J., Puzicha, J.: Shape context: A new descriptor for shape matching and object recognition. Advances in neural information processing systems (2001) 831–837
4. Keysers, D., Deselaers, T., Gollan, C., Ney, H.: Deformation models for image recognition. IEEE Trans on Pattern Analysis and Machine Intelligence **29** (2007) 1422–1435
5. Simard, P., Cun, Y.L., Denker, J., Victorri, B.: Transformation invariance in pattern recognition: Tangent distance and tangent propagation. International Journal of Imaging Systems and Technology **11** (2000) 181–197
6. Hastie, T., Simard, P.: Metrics and models for handwritten character recognition. Statistical Science (1998) 54–65
7. Keysers, D., Macherey, W., Ney, H., Dahmen, J.: Adaptation in statistical pattern recognition using tangent vectors. Pattern Analysis and Machine Intelligence, IEEE Transactions on **26** (2004) 269–274
8. Frey, B., Jojic, N.: Transformed component analysis: Joint estimation of spatial transformations and image components. In: Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on. Volume 2., IEEE (1999) 1190–1196
9. Frey, B., Jojic, N.: Estimating mixture models of images and inferring spatial transformations using the em algorithm. In: Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on. Volume 1., IEEE (1999)
10. Frey, B., Jojic, N.: Transformation-invariant clustering using the em algorithm. Pattern Analysis and Machine Intelligence, IEEE Transactions on **25** (2003) 1–17
11. Miller, E., Matsakis, N., Viola, P.: Learning from one example through shared densities on transforms. In: Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on. Volume 1., IEEE (2000) 464–471
12. Cootes, T., Edwards, G., Taylor, C.: Active appearance models. Pattern Analysis and Machine Intelligence, IEEE Transactions on **23** (2001) 681–685
13. Jones, M., Poggio, T.: Multidimensional morphable models: A framework for representing and matching object classes. International Journal of Computer Vision **29** (1998) 107–131
14. Bruhn, A., Weickert, J., Schnörr, C.: Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. International Journal of Computer Vision **61** (2005) 211–231
15. LeCun, Y.: Mnist handwritten digit database. AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist (1998)